# CONTACT DYNAMICS AND AUTONOMOUS CONTROL DURING RENDEZVOUS AND BERTHING MANEUVERS

**Simone Asci and Angadh Nanjangud**

*Queen Mary University of London, QMUL, London, United Kingdom,*
*s.asci/a.nanjangud@qmul.ac.uk*

## ABSTRACT

Autonomous space robots capable of object manipulation will play a prominent role in future space missions to achieve in-orbit assembly and debris capture. Modeling the robotic manipulation of cooperative or non-cooperative space targets requires a deeper understanding of the interaction dynamics between multibody systems, but, unlike terrestrial robotics, is deficient in real-world experimental data of in-orbit multibody contact dynamics models. Thus computer simulations remain an indispensable tool for spacecraft dynamics and control engineers. The authors present a contact-inclusive simulation framework to demonstrate spacecraft dynamics before, during, and after berthing operations. They also discuss the trade-offs of various approaches to modelling and simulation when ignoring the contact dynamics.

## 1. INTRODUCTION

Autonomous rendezvous and capture/berthing is needed in a variety of scenarios relevant to in-space assembly, servicing, debris capture, and deorbit. Understanding the contact dynamics in these context is essential for safely manipulating targets (cooperative or uncooperative). The difficulty of reproducing the on-orbit environment with ground-based experiments makes it particularly important to numerically simulate such proximity operations in a realistic and accurate way at a lower cost. Specifically, simulating autonomous rendezvous and berthing requires a framework capable of developing and investigating optimal control strategies while replicating contact dynamics with high fidelity. Although several researchers have addressed contact problems during capture, contact dynamics is also relevant in examining post-capture behaviours [1]. To demonstrate such capabilities of the proposed framework, this study examines the scenario of a target body being approached by a chaser, which comprises an actuated base-spacecraft with a four-link manipulator. Overall, the simulation presented in this paper shows result from three phases of the berthing maneuver: pre-capture optimal trajectory execution to a predetermine point on the target; capture-related contact dynamics based on collision detection; and post-berthing behaviours.

The first trajectory optimization problem consists of finding a path in the joint space for the manipulator so that the end-effector (EE) reaches the designated position for the coupling operation by minimizing a prescribed objective function while respecting path bounds and constraints on joint velocity and acceleration. In regards to the second phase of simulating contact during coupling operation, we formulate collision detection as a convex optimization problem [2] which avoids using computationally expensive mesh-based representations of the contact surfaces [3]. Here, we employ an elasto-plastic force model previously used to study space robots [4].In the third phase of the simulation, constraint forces resulting from the interaction between the target and the final link of the robot arm are solved via the use of Lagrange multipliers (LMs) [5].The Lagrange multiplier method allows for the construction of a detachable joint between bodies that make and break contact; it offers a more realistic physical model to treat bodies as separable entities that does not require reformulating system dynamics. A final focus of the paper is the investigation of the behavior of such a detachable joint and its effects on the coupled dynamics of the integrated spacecraft and target during the berthing maneuver. Previous works using Lagrange multipliers derive constraint equations of multibody systems using only external loads [6]; in this work, we add the contact forces generated by collision detection into this computation.

The paper is organized as follows: Section 2 describes the system under consideration and the maneuver relevant details. In Section 3 the problem of optimal trajectory generation is posed, and the algorithm for solving it is described. Section 4 deals with contact dynamics, where the collision detection algorithm and collision resolution formula are described. Section 5 explains the Lagrange multipliers method applied to model the detachable joint. It also differentiates between two approaches considering first the accuracy of motion behaviours at the joint, and the computational effort of simulations. Section 6 presents relevant results and the main conclusions are summarised in Section 7, .

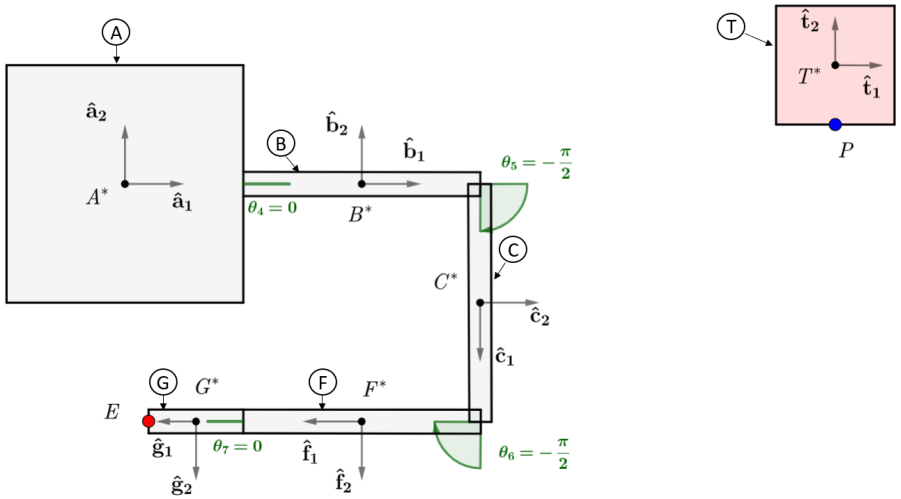


Figure 1: Planar view of the spacecraft and target.

## 2. SYSTEM MODELLING

Figure 1 shows a planar schematic of a multibody chaser vehicle and a single-body target $T$. The chaser comprises a base-spacecraft vehicle $A$ and a four-link manipulator whose links are labelled $B, C, F$, and $G$, from base link to tip link. The following assumptions are made in the kinematic and dynamic modeling of the system:

- All bodies are rigid cuboids, homogeneous, and isotropic; their centers of mass coincide with their respective geometric centers.
- Although the base-spacecraft and the target are modeled with 6 Degrees-of-Freedoms (DoFs), the berthing operation is performed entirely with planar motion.
- dimensions, position, and orientation of the target are known

The base-spacecraft is equipped with a three-axis reaction wheel to change its orientation, which is defined by Euler angles condensed in vector $\boldsymbol{\theta_a} = [\theta_1, \theta_2, \theta_3]$; obviously, there is no need for nozzle jets to move the base-spacecraft center of mass (CoM). The manipulator has four joints defined by as many angles collected in the vector $\boldsymbol{\theta_m} = [\theta_4, \theta_5, \theta_6, \theta_7]$, each joint provides one rotational DoF and is actuated by one motor. The target $T$ is free to translate and rotate in space but lacks onboard actuators, thus it can be moved only by an external interaction. Identical to prior work [7], the target is assumed to be within the grasping range of the manipulator. The multibody dynamics models derived in this study are based on Kane's method [8, 9]. $\mathbf{q}$ denotes the generalized coordinates vector of the chaser, containing the position and attitude of the base-spacecraft $\mathbf{x_a}$ and the joint angles of the manipulator $\theta_\mathbf{m}$, the resulting equations of motion (EoMs) can be succinctly presented as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} \tag{1}$$

where $\mathbf{M}(\mathbf{q})$ is the inertia matrix of the system, $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ is the nonlinear velocity-dependent vector and $\boldsymbol{\tau}$ is the vector of generalized forces and torques. Eq. (1) can be partitioned in the following form:

$$\begin{bmatrix} \mathbf{M_a} & \mathbf{M_{am}} \\ \mathbf{M_{am}}^T & \mathbf{M_m} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{x}}_\mathbf{a} \\ \ddot{\boldsymbol{\theta}}_\boldsymbol{m} \end{bmatrix} + \begin{bmatrix} \mathbf{c_a} \\ \mathbf{c_m} \end{bmatrix} = \begin{bmatrix} \mathbf{f_a} \\ \boldsymbol{\tau_m} \end{bmatrix} + \begin{bmatrix} \mathbf{J_a}^T \\ \mathbf{J_m}^T \end{bmatrix} \mathbf{f_e} \tag{2}$$

where $\mathbf{M_a} \in \mathbb{R}^{6\times6}$ is the base-spacecraft inertia matrix; $\mathbf{M_m} \in \mathbb{R}^{n\times n}$ is the manipulator inertia matrix; $\mathbf{M_{am}} \in \mathbb{R}^{6\times n}$ is the coupled inertia matrix; $\ddot{\mathbf{x}}_\mathbf{a} \in \mathbb{R}^6$ is the linear and angular accelerations vector of the base-spacecraft, i.e., second derivative of $\mathbf{x_a}$; $\ddot{\boldsymbol{\theta}}_\boldsymbol{m} \in \mathbb{R}^n$ is the vector of angular accelerations of the manipulator, i.e., second derivative of $\boldsymbol{\theta_m}$; $\mathbf{c_a} \in \mathbb{R}^6$ and $\mathbf{c_m} \in \mathbb{R}^n$ are the nonlinear velocity-dependent vectors for the base-spacecraft and manipulator, respectively; $\mathbf{f_a} \in \mathbb{R}^6$ and $\mathbf{f_e} \in \mathbb{R}^6$ are the vectors of external forces and moments on the base-spacecraft CoM and the EE of the manipulator, respectively; while $\boldsymbol{\tau_m} \in \mathbb{R}^n$ is the vector of joint torques of the manipulator; $\mathbf{J_a} \in \mathbb{R}^{6\times6}$ and $\mathbf{J_m} \in \mathbb{R}^{6\times n}$ are the Jacobian matrices with respect to the base-spacecraft and the manipulator, respectively. In this paper, $n = 4$ since the manipulator has four links. Starting from a folded configuration, the manipulator approaches the stationary target with a trajectory that minimizes the angular accelerations. The EE arrives with zero velocity at a point on the target designated for coupling. Then, a hinge constraint is established, unifying spacecraft and target into a new multibody system. Interactions with the target affect the position of the manipulator and, because of coupled dynamics, also the attitude of the base-spacecraft, which can be re-stabilized through the reaction wheel. Since priority must be given to the execution of the coupling operation, which requires precise alignment between bodies, a "computed torque controller" [10] was chosen to track the trajectory, allowing full control of the spacecraft actuation. After the coupling operation is completed, the first part of the berthing maneuver is performed, dragging the target close to the base-spacecraft with constant torque values of joint motors.

## 3. OPTIMAL TRAJECTORY GENERATION

Convex programming techniques are widely used in aerospace guidance and control applications [11]. Specifically in the context of berthing maneuvers, pre-capture trajectories were presented where the base-spacecraft's attitude is controlled so that it maintains the same rotational rate relative to a tumbling target [12]. In this paper, a trajectory optimization algorithm minimizes the sum of base-spacecraft and joints squared accelerations over the examined time horizon; since the base-spacecraft CoM is fixed, only angular coordinates are considered.

$$
\begin{aligned}
\underset{\ddot{\theta}_{4,i},\ddot{\theta}_{5,i},\ddot{\theta}_{6,i},\ddot{\theta}_{7,i}}{\text{minimize}} \quad & \sum_{i=1}^{N}\sum_{j=4}^{7}\ddot{\theta}_{j,i}^2 + \sum_{i=1}^{N}\sum_{j=1}^{3}\alpha_{j,i}^2 \\
\text{subject to} \quad & -\mathbf{L_1} \leq \boldsymbol{\theta_i} \leq \mathbf{L_1}, \\
& -\mathbf{L_2} \leq \dot{\boldsymbol{\theta}}_{\boldsymbol{i}} \leq \mathbf{L_2}, \\
& -\mathbf{A_{max}} \leq \ddot{\boldsymbol{\theta}}_{\boldsymbol{m,i}} \leq \mathbf{A_{max}}, \\
& -\boldsymbol{\alpha_{max}} \leq \boldsymbol{\alpha_i} \leq \boldsymbol{\alpha_{max}}, \\
& \boldsymbol{\theta(0)} = \boldsymbol{\theta_0}, \\
& \boldsymbol{\theta(T)} = \boldsymbol{\theta_T}, \\
& \dot{\boldsymbol{\theta}}(\mathbf{0}) = \dot{\boldsymbol{\theta}}_{\mathbf{0}}, \\
& \dot{\boldsymbol{\theta}}(\boldsymbol{T}) = \dot{\boldsymbol{\theta}}_{\boldsymbol{T}}, \\
& \ddot{\boldsymbol{\theta}}_{\boldsymbol{a,i}} = f(\boldsymbol{\alpha_i}), \\
& 0 \leq t \leq T
\end{aligned}
\tag{3}
$$

where the joint accelerations $\ddot{\theta}_{j,i},\ j = (4,5,6,7)$, condensed into the vector $\ddot{\boldsymbol{\theta}}_{\boldsymbol{m,i}}$, are the control variables at the specific time instant $i$; the limits to the accelerations are defined by vector $\mathbf{A_{max}}$. $\boldsymbol{\theta_i} = [\boldsymbol{\theta_{a,i}}, \boldsymbol{\theta_{m,i}}]$ is the angular positions of the chaser and $\dot{\boldsymbol{\theta}}_{\boldsymbol{i}} = [\dot{\boldsymbol{\theta}}_{\boldsymbol{a,i}}, \dot{\boldsymbol{\theta}}_{\boldsymbol{m,i}}]$ are the angular speeds of the chaser. They are subjected respectively to path bounds $\mathbf{L_1}, \mathbf{L_2}$. $\boldsymbol{\theta_0}, \dot{\boldsymbol{\theta}}_{\mathbf{0}}$ and $\boldsymbol{\theta_T}, \dot{\boldsymbol{\theta}}_{\boldsymbol{T}}$ are the constraints on the initial and final configuration state of the system respectively. Vector $\boldsymbol{\alpha_i}$ stores angular accelerations of the base-spacecraft for the specific instant $i$, whose limits are defined in the bound vector $\boldsymbol{\alpha_{max}}$. Function $f(\boldsymbol{\alpha})$ expresses the mapping between the base-spacecraft angular accelerations and double derivatives of Euler angles $\ddot{\boldsymbol{\theta}}_{\boldsymbol{a,i}} = (\ddot{\theta}_{1,i}, \ddot{\theta}_{2,i}, \ddot{\theta}_{3,i})$ Finally, parameter $T$ indicates the duration of the maneuver, it is divided into $N$ intervals of $i$ length. The dynamics of the system is managed by the computed torque controller described in Equation 4. The computed torque controller is a nonlinear control technique that combines the benefits given by the dynamic model and the stabilization of a PD controller, it is based on the feedback linearization approach. It is composed of a feedforward component to achieve the nominal required torques and a feedback component to minimize position and velocity tracking errors. Given the spacecraft dynamic model and a twice-differentiable desired trajectory, which is generated by the optimization algorithm, this control law calculates the torque functions for the trajectory execution. The first component of the controller is the product between the inertia matrix of the system $\mathbf{M(q)}$ (introduced in eq. (1)) and an acceleration term which is the sum of the feedforward acceleration $\ddot{\mathbf{q}}_{\mathbf{d}}$ and a feedback acceleration which stabilizes the trajectory error to zero, generated by a PD controller: $\mathbf{K_p} \cdot (\mathbf{q_d} - \mathbf{q}) + \mathbf{K_d} \cdot (\dot{\mathbf{q}}_{\mathbf{d}} - \dot{\mathbf{q}})$. The feedforward plus feedback accelerations compose the commanded acceleration which is turned by the inertia matrix into a joint torque. The second component of the controller $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ is the feedforward term providing the torque to compensate for nonlinearities, such as Coriolis and gravity forces. The formula is shown in Eq. (4):

$$
\begin{aligned}
\boldsymbol{\tau} = \boldsymbol{M(q)} \Big[ \ddot{\boldsymbol{q}}_{\boldsymbol{d}} + \boldsymbol{K_p} \cdot (\boldsymbol{q_d} - \boldsymbol{q}) + \boldsymbol{K_d} \cdot (\dot{\boldsymbol{q}}_{\boldsymbol{d}} - \dot{\boldsymbol{q}}) \Big] \\
+ \boldsymbol{c(q, \dot{q})}
\end{aligned}
\tag{4}
$$

where the desired position $\mathbf{q_d}$, velocity $\dot{\mathbf{q}}_{\mathbf{d}}$ and acceleration $\ddot{\mathbf{q}}_{\mathbf{d}}$ are obtained from trajectory planning; $\mathbf{K_p}$ and $\mathbf{K_d}$ are positive diagonal matrices; $\mathbf{M(q)}$ and $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ matrices are provided by the dynamic model of the spacecraft. It should be considered that this controller requires actuation on the base-spacecraft, therefore the chaser cannot be in a free-floating state; on the other hand, full exploitation of the system's actuators allows precise trajectory tracking.

## 4. CONTACT MODEL

The simulation results presented in this paper are for collisions between the target $T$ and the final link $G$ of the multibody chaser but we have also examined the collision dynamics between $T$ and other bodies of the chaser as well, which was examined in our work on debris collisions [13]. Note, however, that one limitation of our simulation is that it does not currently prevent self-collision between bodies of the chaser spacecraft itself. This allows, for example, the overlapping between links during planar motion. This issue of overlapping of manipulator links in the planar perspective is physically realistic because orbital manipulators often have joint offsets between their links, which facilitates compact stowing. As a result, the links of real-world manipulators are physically designed to never come in contact with each other and do overlap in a planar sense. Of course, the links of manipulator could collide with the base vehicle, this can be handled by our simulation if needed but our focus in this work here is to investigate the effects of contact between bodies that are external to the chaser system.

Our approach to simulating contact dynamics relies on the concept of minimum distance points (MDPs), which in the context of this paper is defined as a pair of points on two bodies (i.e., one point on each body) that are closest to each other. Keeping track of this minimum distance between two points on two disconnected bodies allows the determination of when a collision event occurs, this is formally known as collision detection. The two bodies are said to be in contact if the minimum distance between the two points is zero, in this case, we proceed to compute the contact/collision forces between the two bodies.

Indeed, if the minimum distance between the two points is a non-zero real value, then the two bodies are not in contact. The pair of points that make up the MDPs are always located on the boundaries of the bodies and are the points where a contact event/collision occurs in our model. Knowledge of the coordinates of the MDP pair allows us to compute the interpenetration between bodies, represented by $\rho$ (in SI units of $m$); this is defined as the depth of penetration between bodies that are, in fact, in a collision state. Interpenetration is then used to determine the direction of the reaction force as well as its magnitude.

From a mathematical perspective, we formulate this collision detection part of the simulation as an inequality-constrained convex optimization program that performs a minimization over a continuous objective function. In this problem, the objective function represents a scalar metric describing the squared minimum distance between two bodies, referred to as proximity and represented by the symbol $\phi$ (SI units of m). In terms of the paragraph immediately above, $\phi$ defines the minimum distance between two points (i.e., the MDPs-pair), which we now know can be either zero or non-zero positive scalar. So, if $\phi$ is zero, the bodies are in contact and the contact portion of the simulation can then be performed. Figure 2 shows $\phi$ as the distance between points $\tilde{X}$ and $\tilde{Y}$, these two points are the MDPs of bodies $X$ and $Y$ respectively. Note that this figure shows $X$ and $Y$ as cuboids with half-lengths of their sides defined by $C_i$ and $D_i$ $(i = 1, 2, 3)$, respectively. Bodies $X$ and $Y$ in Figure 2 are useful in describing the collision detection problem but the reader should also see them as analogs for bodies $G$ and $T$ in Figure 1. Further, in Figure 2, we highlight two position vectors that track $\tilde{X}$ and $\tilde{Y}$ from the origin of an external frame $I$; these are indicated as $\tilde{\mathbf{r}}_\mathbf{X}$ and $\tilde{\mathbf{r}}_\mathbf{Y}$, respectively. These serve as the independent variables of our convex optimisation problem and their optimal values provide the coordinates of the MDPs-pair. In other words, their optimal value is that which minimizes the objective function representing the distance between $\tilde{X}$ and $\tilde{Y}$ to zero. The coordinates of each point are used to determine the amount of interpenetration $\rho$, which is later used to compute the reaction force.

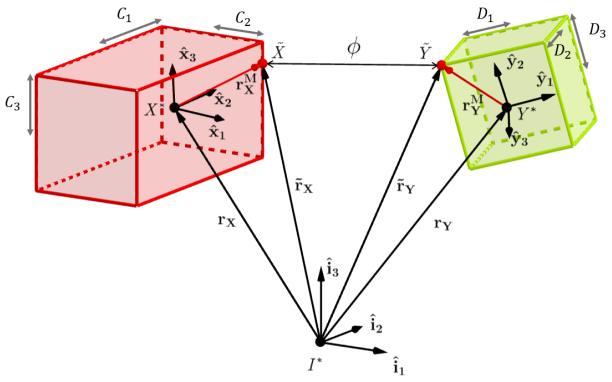


Figure 2: Geometrical description of two cuboids as defined by their centroids, orthogonal basis and parameters.

Note that the optimisation problem can only be correctly solved (i.e. minimizing $\phi$ for the MDPs coordinates) when the objects do not intersect or overlap. This is due to the fact that every pair of points within the overlapping region are valid solutions to MDP-pair that also minimize the objective function to zero but no longer represent the original points where contact occurs in reality. Thus, to correctly solve the optimisation for the case of overlapping bodies, we introduce a fictitious body to compute $\rho$ that helps correct our simulation; this is described in greater detail below.

We formulate a new problem that makes it possible to calculate $\rho$ by introducing a fictitious cuboid at the same location as the secondary body (shown in green in Figure 2). This fictitious body has smaller set of dimensions than the secondary body; we indicate this fictitious body dimensions by $D_i^*$ $(i = 1, 2, 3)$, which is given by:

$$D_i^* = D_i - b. \tag{5}$$

where $b$ is a positive parameter. A surrogate proximity, $\phi^* \triangleq \|\tilde{\mathbf{r}}_\mathbf{X} - \tilde{\mathbf{r}}_\mathbf{Y}^*\|^2$, can be computed by solving the following inequality constrained convex program:

$$\begin{aligned} \underset{\tilde{\mathbf{r}}_\mathbf{X}, \tilde{\mathbf{r}}_\mathbf{Y}^*}{\text{minimize}} \quad & \|\tilde{\mathbf{r}}_\mathbf{X} - \tilde{\mathbf{r}}_\mathbf{Y}^*\|^2 \\ \text{subject to} \quad & |\, \tilde{\mathbf{r}}_\mathbf{X} \cdot \hat{\mathbf{x}}_\mathbf{1} \,| \leq C_1, \\ & |\, \tilde{\mathbf{r}}_\mathbf{X} \cdot \hat{\mathbf{x}}_\mathbf{2} \,| \leq C_2, \\ & |\, \tilde{\mathbf{r}}_\mathbf{X} \cdot \hat{\mathbf{x}}_\mathbf{3} \,| \leq C_3, \\ & |\, \tilde{\mathbf{r}}_\mathbf{Y}^* \cdot \hat{\mathbf{y}}_\mathbf{1} \,| \leq D_1^*, \\ & |\, \tilde{\mathbf{r}}_\mathbf{Y}^* \cdot \hat{\mathbf{y}}_\mathbf{2} \,| \leq D_2^*, \\ & |\, \tilde{\mathbf{r}}_\mathbf{Y}^* \cdot \hat{\mathbf{y}}_\mathbf{3} \,| \leq D_3^* \end{aligned} \tag{6}$$

where $\tilde{\mathbf{r}}_\mathbf{X}$ and $\tilde{\mathbf{r}}_\mathbf{Y}^*$ are the positions of the MDPs belonging to the primary cuboid and the fictitious cuboid respectively from the origin of the inertial reference frame. The interpenetration between the original bodies $\rho$, can then be recovered from $\phi^*$ as shown below:

$$\rho = \begin{cases} 0 & \text{for } \phi^* > b \\ b - \phi^* & \text{for } 0 < \phi^* \leq b \end{cases} \tag{7}$$

This is sufficient to correctly detect collisions but simulating contact dynamics requires determination of the contact normal to the surfaces, directed towards the body and parallel to the collision force. $\hat{\mathbf{n}}_\mathbf{X}$ denotes this surface normal for the primary cuboid and is computed from Eq. (8):

$$\hat{\mathbf{n}}_\mathbf{X} = \frac{\tilde{\mathbf{r}}_\mathbf{Y} - \tilde{\mathbf{r}}_\mathbf{X}}{\|\tilde{\mathbf{r}}_\mathbf{Y} - \tilde{\mathbf{r}}_\mathbf{X}\|} \tag{8}$$

The collision detection work described in this section solves for $\tilde{\mathbf{r}}_\mathbf{X}$ in Equation 6; we can also compute $\tilde{\mathbf{r}}_\mathbf{Y}$ with its support using

$$\tilde{\mathbf{r}}_\mathbf{Y} \approx \left( \tilde{\mathbf{r}}_\mathbf{Y}^* + b \frac{\tilde{\mathbf{r}}_\mathbf{X}^* - \tilde{\mathbf{r}}_\mathbf{Y}^*}{\|\tilde{\mathbf{r}}_\mathbf{X}^* - \tilde{\mathbf{r}}_\mathbf{Y}^*\|} \right) \tag{9}$$

For non-zero $\rho$ and knowledge of $\hat{\mathbf{n}}_\mathbf{X}$, we can now compute the contact force at a realistic contact point. For example, $\mathbf{F}_{\tilde{\mathbf{X}}}$, the contact force at $\tilde{X}$, can be computed from

$$\mathbf{F}_{\tilde{\mathbf{X}}} = (k_c \cdot \rho^3) \cdot (1 - c_c \cdot \dot{\rho}) \cdot \hat{\mathbf{n}}_\mathbf{X} \tag{10}$$

where $k_c$ and $c_c$ are the contact stiffness and damping coefficient in the normal direction to the contact surface, respectively. In this paper, the contact forces are defined only in the normal direction to the contact surface and contribution of the tangential component is assumed to be negligible. This simplified elastic-plastic contact force model [14] (also referred to as penalty-based method) defining the interaction between colliding bodies is composed of an elastic component proportional to the virtual interpenetration $\rho$, and a plastic component which is related to $\dot{\rho}$. Thus it is essentially a spring-damper model and requires complete information about the materials, geometries, and velocities of the bodies involved.

This contact force acts on a contact point but can be replaced by a force acting through the centre of mass, $\mathbf{F_X}$, and a moment about the mass centre $\mathbf{M_X}$, which are given by:

$$\begin{aligned} \mathbf{F_X} &= \mathbf{F_{\tilde{X}}} \\ \mathbf{M_X} &= \mathbf{r_X^M} \times \mathbf{F_{\tilde{X}}}, \end{aligned} \tag{11}$$

where $\mathbf{r_X^M}$, the position vector from the mass centre of the primary cuboid to $\tilde{X}$, is determined from

$$\mathbf{r_X^M} = \tilde{\mathbf{r}}_\mathbf{X} - \mathbf{r_X} \tag{12}$$

A force in the opposite direction acts on the CoM of the secondary cuboid, and the relative torque can be computed accordingly based on the position vector $\mathbf{r_Y^M}$ of contact point $\tilde{Y}$ from center $Y^*$. This is not repeated here for sake of brevity.

## 5. LAGRANGE MULTIPLIERS BASED CONSTRAINT

The detachable joint between the target and the last link of the manipulator is modeled through the method of LMs. This approach is based on absolute coordinates formulation, which means that: the bodies involved must be treated as independent and the dynamic model contains sparse equations compared to that obtained by a minimal coordinates approach. On the other hand, it is simple to set up and is not plagued by the limitation of generating immutable dynamic models, meaning that it is possible to detach and reattach bodies at runtime, without the need to derive a dynamic model from scratch for every new system, as required instead by the minimum coordinate formulation. The detachable joint is modeled as a hinge, that is, as a bilateral constraint that prevents the relative translation of the bodies but allows their relative rotation. The last link of the manipulator belongs to a compound system (the base-spacecraft and three other links of the manipulator) and thus is replaced with a new, fictitious single body, with equivalent kinematics and inertial properties. The fictitious body (referred to as "dummy link") is used to compute the constraints forces, which will be then applied to the original body. Specifically, with reference to Figure 1, the constraint is established between point $P$ on the target and the EE point $E$, and the latter is replaced by a surrogate point on the dummy link denoted by $E^*$.

The Lagrange multipliers method is therefore applied to two bodies subject to one constraint, forming a new dynamic system; in this section, all terms referring to such a system are indicated with the suffix *L*. The positions of both bodies are lumped into a single state vector, $\mathbf{q_L}$, which has a length of 12 because every single body in 3D space is characterized by 6 coordinates. The inertia matrix of the system, indicated as $\mathbf{M_L}$, collects the masses and inertia of both bodies as diagonal elements, while off-diagonal elements are zero. Grouping all the forces of the system in a global force vector $\tilde{\mathbf{Q}}$, the equation of motion is:

$$\ddot{\mathbf{q}}_\mathbf{L} = \mathbf{W_L}\tilde{\mathbf{Q}} \tag{13}$$

where $\mathbf{W_L}$ is the inverse of the mass matrix. Since the bodies are subject to a single constraint $\mathbf{C}$, this is expressed as a one-element vector as a function of $\mathbf{q_L}$:

$$\mathbf{C}(\mathbf{q_L}) = \left[\frac{1}{2}(\mathbf{x}(\mathbf{q_L}) \cdot \mathbf{x}(\mathbf{q_L}) - l)\right] \tag{14}$$

where vector $\mathbf{x}$ expresses the distance between $P$ and $E^*$ and $l$ is the desired value for such distance, which is typically set to zero for a hinge constraint. Positions of coupling points consistent with the constraint are those that satisfy:

$$\mathbf{C} = 0 \tag{15}$$

Consistent velocities are those satisfying Eq. 16:

$$\dot{\mathbf{C}} = 0 \tag{16}$$

Similarly, consistent accelerations need to satisfy Eq. 17:

$$\ddot{\mathbf{C}} = 0 \tag{17}$$

Derivatives of $\mathbf{C}$ are obtained by applying the chain rule:

$$\dot{\mathbf{C}} = \frac{\partial \mathbf{C}}{\partial \mathbf{q_L}}\dot{\mathbf{q}}_\mathbf{L} = \mathbf{J}\dot{\mathbf{q}}_\mathbf{L} \tag{18}$$

where $\mathbf{J}$ indicates the Jacobian matrix of $\mathbf{C}$ with respect to the state vector $\mathbf{q_L}$: $\mathbf{J} = \partial\mathbf{C}/\partial\mathbf{q_L}$. Differentiating again with respect to time gives Eq. 19:

$$\ddot{\mathbf{C}} = \dot{\mathbf{J}}\dot{\mathbf{q}}_\mathbf{L} + \mathbf{J}\ddot{\mathbf{q}}_\mathbf{L} \tag{19}$$

Therefore, the problem consists of solving for a constraint force that, added to the external force ensures that the resulting acceleration complies with Eq. 17, which means that the constraint force must cancel the component of the external force pointing out of the allowed direction of motion. It follows that the constraint force is orthogonal to the allowed motion and therefore does no work, in agreement with the principle of virtual work [15], according to which constraint forces neither add nor take away energy from the system. Substituting Eq. (13) in (19) gives:

$$\ddot{\mathbf{C}} = \dot{\mathbf{J}}\dot{\mathbf{q}}_\mathbf{L} + \mathbf{J}\mathbf{W_L}\tilde{\mathbf{Q}} \tag{20}$$

The global force vector $\tilde{\mathbf{Q}}$ can be partitioned in the vector of external forces $\mathbf{Q}$ and vector of constraint forces $\hat{\mathbf{Q}}$ as follows: $\tilde{\mathbf{Q}} = \mathbf{Q} + \hat{\mathbf{Q}}$. Setting to zero according to Eq. (17) and re-arranging gives:

$$\mathbf{J}\mathbf{W}_\mathbf{L}\hat{\mathbf{Q}} = -\dot{\mathbf{J}}\dot{\mathbf{q}}_\mathbf{L} - \mathbf{J}\mathbf{W}_\mathbf{L}\mathbf{Q} \tag{21}$$

To ensure that the constraint forces do no work, they must comply with the following equation:

$$\hat{\mathbf{Q}} \cdot \dot{\mathbf{x}}(\mathbf{q_L}) = 0, \quad \forall \dot{\mathbf{x}}(\mathbf{q_L}) \mid \mathbf{J}\dot{\mathbf{x}}(\mathbf{q_L}) = 0 \tag{22}$$

and from this it follows that vector $\hat{\mathbf{Q}}$ can be expressed as:

$$\hat{\mathbf{Q}} = \mathbf{J}^\mathbf{T}\boldsymbol{\lambda} \tag{23}$$

where $\boldsymbol{\lambda}$ is a vector with the dimension of $\mathbf{C}$ whose components are known as *Lagrange multipliers*. By replacing Eq. 23 in (21):

$$\mathbf{J}\mathbf{W}_\mathbf{L}\mathbf{J}^\mathbf{T}\boldsymbol{\lambda} = -\dot{\mathbf{J}}\dot{\mathbf{q}} - \mathbf{J}\mathbf{W}_\mathbf{L}\mathbf{Q} \tag{24}$$

from which $\boldsymbol{\lambda}$ can be obtained and multiplied by $\mathbf{J}^\mathbf{T}$ to give $\hat{\mathbf{Q}}$, which is then added to the vector of acting forces before calculating the accelerations. Since the numerical solution of the EoMs is plagued by drift, a feedback term is required to prevent the accumulation of numerical drift; the feedback term is usually modeled as a damped-spring force and is incorporated directly into the constraint force calculation as follows:

$$\mathbf{J}\mathbf{W}_\mathbf{L}\mathbf{J}^\mathbf{T}\boldsymbol{\lambda} = -\dot{\mathbf{J}}\dot{\mathbf{q}} - \mathbf{J}\mathbf{W}_\mathbf{L}\mathbf{Q} - K_s^\lambda\mathbf{C} - K_d^\lambda\dot{\mathbf{C}} \tag{25}$$

where $K_s^\lambda$ and $K_d^\lambda$ are gains that reproduce the spring and damping constants. By adding the feedback term, compliance with the constraint is further enforced by canceling out any displacement caused by drift, thus returning the system to a valid state. Thus, during the simulation the relative distance of the bodies is not always maintained at the desired value, but there is a continuous alternation of drift and error compensation, which generates an oscillating trend of such distance. Therefore the role of appropriately compensating for the accumulation of the integration error is relevant; specifically, for the hinge, it is also particularly demanding because any displacement that alters the original distance of the coupling points in any direction must be compensated for. The basic approach implemented with Eq. 25 has two drawbacks:

- gains are weighted with $\mathbf{C}$ and $\dot{\mathbf{C}}$, that is, with the current relative distance and velocity of the contact points; however, the adaptivity of the resulting feedback terms is characterized by low responsiveness because based on the current state of the constraint instead of the forces that were previously responsible for it.
- gains are usually assigned arbitrary constant values, chosen through a trial-and-error approach, while they should vary according to the current state of the constraint to ensure better performance and mitigate instabilities.

This can cause poor performance when the dynamics of the system changes abruptly; in fact, undersized restoring forces cause uncontrollable growth of constraint drift. On the other hand, responding with strong restoring forces to small displacements gives rise to stiff differential equations that are intractable. A solution is offered by enabling collision detection inside the integrated system, i.e. between the joined bodies. This makes it possible to use internal reaction forces to determine whether the bodies are in contact (i.e., the constraint is satisfied) or not (i.e., the bodies are moving away and the constraint is about to be violated), this signal is endowed with greater responsiveness than simply measuring the actual distance between bodies, because the forces are responsible for the accelerations and thus the displacements of the bodies in the following steps. According to this approach, gains can be defined as follows:

$$\begin{cases} K_s^\lambda, K_d^\lambda = \alpha_1, \beta_1 & \text{if } \|F_{\tilde{A}}\| > \gamma \\ K_s^\lambda, K_d^\lambda = \alpha_2, \beta_2 & \text{if } \|F_{\tilde{A}}\| \leq \gamma \end{cases} \tag{26}$$

where $F_{\tilde{A}}$ is the contact force between the bodies, $\alpha_i, \beta_i,\ i = (1, 2, 3)$ and $\gamma$ are constants. A comparison has been carried out between following scenarios:

1. CD-disabled: collision detection disabled, LMs with constant gains
2. CD-enabled: collision detection enabled, LMs with adaptive gains

## 6. RESULTS

Due to computational limitations, the time of the simulation is not to scale. It has a total duration of 7 seconds, the first 5 of which are employed for the optimal trajectory execution and the last 2 for the post-coupling maneuver. The coupling operation takes place between the two phases, in a fast and direct manner. The system dynamics parameters are summarized in Table 1, where $l_i, w_i, d_i, i = (A, B, C, F, G, T)$ indicate the

Table 1: **Dynamics parameters**

| Body | $l_i$ [m] | $w_i$ [m] | $d_i$ [m] | $m_i$ [kg] | $I^i_{xx}$ [kg m$^2$] | $I^i_{yy}$ [kg m$^2$] | $I^i_{zz}$ [kg m$^2$] |
|---|---|---|---|---|---|---|---|
| Base A | 0.5 | 0.5 | 0.5 | 20 | 0.833 | 0.83 | 0.83 |
| Link B | 0.5 | 0.05 | 0.05 | 2 | 0.0008 | 0.042 | 0.042 |
| Link C | 0.5 | 0.05 | 0.05 | 2 | 0.0008 | 0.042 | 0.042 |
| Link F | 0.5 | 0.05 | 0.05 | 2 | 0.0008 | 0.042 | 0.042 |
| Link G | 0.1 | 0.05 | 0.05 | 2 | 0.0008 | 0.02 | 0.02 |
| Target T | 0.2 | 0.2 | 0.2 | 1 | 0.0067 | 0.0067 | 0.0067 |

length, width and depth of each cuboid. The kinematics boundaries for the optimal trajectory are shown in Table 2. Here $x_A, y_A$ and $x_E, y_E$ are the coordinates of the base-spacecraft CoM and the EE respectively, while $\theta_i, i = (1, .., 7)$ indicate, in order, the angles of base-spacecraft attitude and manipulator joints. Figure 3

Table 2: **Initial and final kinematic boundaries**

| | $x_A$ | $y_A$ | $\theta_1$ | $\theta_2$ | $\theta_3$ | $\theta_4$ | $\theta_5$ | $\theta_6$ | $\theta_7$ | $x_E$ | $y_E$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Position $t_i$ | 0 | 0 | 0 | 0 | 0 | 0 | -$\pi/2$ | -$\pi/2$ | 0 | 0.3 | -1 |
| Position $t_f$ | 0 | 0 | 0 | 0 | 0 | -$\pi/6$ | $\pi/10$ | $\pi/3$ | $\pi/4$ | 3 | 0.23 |
| Velocity $t_i$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Velocity $t_f$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

shows the trajectory of the EE, which in the first phase of the simulation is guided by the controller to follow the optimal trajectory and reach the target. Standing on the CoM of the base-spacecraft, which coincides with the origin of the cartesian plane, the actual trajectory of the EE point, highlighted in blue, tracks the optimal one (in orange) until the designated coupling point on the target is reached (the red dot), and then it describes another portion of trajectory that corresponds to the post-capture maneuver, which, however, is not optimized. The detach-

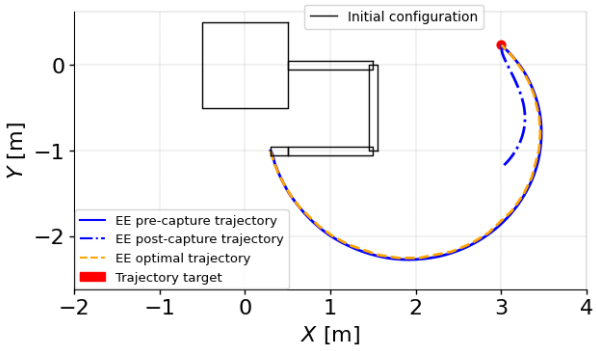


Figure 3: Pre-capture and post-capture EE trajectories seen from the base-spacecraft CoM. The optimal reference trajectory is highlighted in orange.

able hinge is formed after the coupling operation at the fifth second and is maintained until the end of the simulation. Figure 4 compares the behavior of the hinge in the aforementioned scenarios, i.e. with collision detection (CD)-disabled and CD-enabled, depicting the distance between the coupling points. After an initial touch, the points $E$ and $P$ are kept almost constantly at a distance of $0.08$ m despite the desired constraint distance being set to zero. The behavior of the hinge in the two scenarios begins to diverge from about $6.5$ s, when a growing constraint violation in the CD-disabled scenario causes the joint to collapse. Thus, the CD-enabled scenario performs in a more physically realistic way, nevertheless, it requires maintaining compliance with the constraint in the presence of contact forces between the coupling bodies, which could result in the sudden generation of high accelerations, leading to instability.

Figures 5 and 6 compare the system configurations for both scenarios. The first phases of the simulation are identical, while the post-coupling maneuvers differ due to whether or not CD is enabled between the manipulator and the target. While in the first scenario the trajectory of target CoM is very smooth, the second shows a better reproduction of the real behavior of the spacecraft system

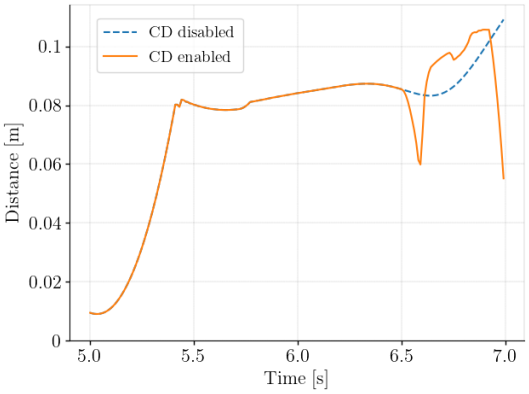


Figure 4: Distance between coupling points $E$ and $P$ in CD-disabled and CD-enabled scenarios

during a capture operation, which, like every other on-orbit proximity operation, is plagued by errors and imperfections. To put this behavior more prominently, in this simulation the connection between the bodies for the berthing operation is hinged rather than fixed.

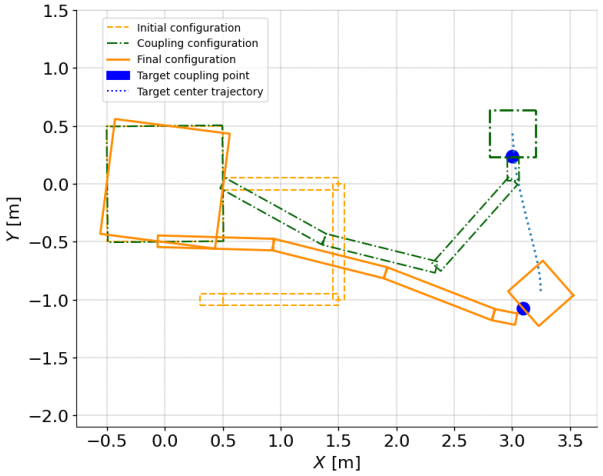


Figure 5: Initial, coupling and final configurations of the system. CD-disabled scenario.

## 7. CONCLUSIONS

This work investigates the interaction between a chaser spacecraft and a target during the simulation of a berthing maneuver. The manipulator tracks a trajectory that minimizes angular accelerations of the base and approaches the stationary target with a final velocity of zero, so as not to cause disturbance in the target pose, which by assumption is not provided with self-actuation. The proposed approach to model a detachable joint exploits the contact forces between joined bodies; the forces are computed by enabling collision detection even after the coupling operation has been performed. This method, verified in sim-

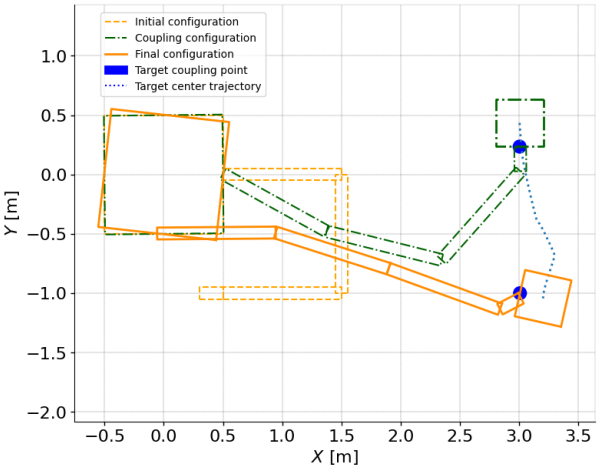


Figure 6: Initial, coupling and final configurations of the system. CD-enabled scenario.

ulation, is in contrast to the standard approach which ignores internal reaction forces between joined bodies. A key contribution of this work is to demonstrate the potential effectiveness of the contact method in reproducing the errors that usually occur during coupling between bodies, making the simulation more realistic; however, this comes at the expense of the additional computational effort required. Further, by providing insights on how to increase the simulation accuracy when an external body is attached to a space manipulator, the results of this study are expected to contribute to the design of post-capture controllers.

## REFERENCES


[1] Shuang Li and Yuchen She. Recent advances in contact dynamics and post-capture control for combined spacecraft. *Progress in Aerospace Sciences*, 120:100678, 2021.

[2] Simone Asci and Angadh Nanjangud. Leveraging symbolic algebra systems to simulate contact dynamics in rigid body systems. *submitted for publication*, 2023.

[3] Gerhard Hippmann. Polygonal contact model revisited: notes on usage and improved implementation. *Multibody System Dynamics*, pages 1–13, 2023.

[4] Jakub Oleś, Tomasz Rybus, Karol Seweryn, Marek Surowiec, Marek Wojtyra, Markus Pietras, and Marc Scheper. Testing and simulation of contact during on-orbit operations. In *Proceedings of 14th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA'2017)*. European Space Agency, 2017.

[5] Edward J Haug. *Computer aided kinematics and dynamics of mechanical systems*, volume 1. Allyn and Bacon Boston, 1989.

[6] Carmine M Pappalardo and Domenico Guida. On the lagrange multipliers of the intrinsic constraint equations of rigid multibody mechanical systems. *Archive of Applied Mechanics*, 88:419–451, 2018.

[7] Angel Flores-Abad, Ou Ma, Khanh Pham, and Steve Ulrich. A review of space robotics technologies for on-orbit servicing. *Progress in aerospace sciences*, 68:1–26, 2014.

[8] Dane Rosenthal and Michael A Sherman. High performance multibody simulations via symbolic equation manipulation and kane's method. *Journal of the Astronautical Sciences*, 34(3):223–239, 1986.

[9] Thomas R Kane, Peter W Likins, and David A Levinson. Spacecraft dynamics. *New York*, 1983.

[10] Chae An, Christopher Atkeson, John Griffiths, and John Hollerbach. Experimental evaluation of feedforward and computed torque control. In *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, volume 4, pages 165–168. IEEE, 1987.

[11] Evangelos Papadopoulos, Farhad Aghili, Ou Ma, and Roberto Lampariello. Robotic manipulation and capture in space: A survey. *Frontiers in Robotics and AI*, 8:228, 2021.

[12] Farhad Aghili. Optimal control of a space manipulator for detumbling of a target satellite. In *2009 IEEE international conference on robotics and automation*, pages 3019–3024. IEEE, 2009.

[13] Simone Asci and Angadh Nanjangud. Towards a generalizable simulation framework to study collisions between spacecraft and debris. In *2022 space flight mechanics meeting*, page 705, 2022.

[14] Gianni Gilardi and Inna Sharf. Literature survey of contact dynamics modelling. *Mechanism and machine theory*, 37(10):1213–1239, 2002.

[15] Herbert Goldstein, Charles Poole, and John Safko. Classical mechanics, 2002.